
Subject: Re: Technique to find maximum in 100x100 element moving box
Posted by [Markus Schmassmann](#) on Thu, 13 Oct 2016 10:36:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 10/12/2016 11:26 PM, Samantha Tushaus wrote:

> For reference, this is my current code in which I essentially use a truncated edge method.
>

```
> FOR i = 0, nx-1 DO BEGIN
>   FOR j = 0, ny-1 DO BEGIN
>     IF (i-50) LT 0 THEN low_ind_i = 0 ELSE low_ind_i = i-50
>     IF (j-50) LT 0 THEN low_ind_j = 0 ELSE low_ind_j = j-50
>     IF (i+50) GT (nx-1) THEN hi_ind_i = nx-1 ELSE hi_ind_i = i+50
>     IF (j+50) GT (ny-1) THEN hi_ind_j = ny-1 ELSE hi_ind_j = j+50
>     data_max[i,j] = max(data[low_ind_i:hi_ind_i,low_ind_j:hi_ind_j])
>   ENDFOR
> ENDFOR
```

still looping, but faster without the ifs:

```
data_max=make_array(size(data,/dim),type=size(data,/type))
FOR i = 0, nx-1 DO FOR j = 0, ny-1 DO data_max[i,j] = $
    max(data[(i-50)>0:(i+50)<nx-1,(j-50)>0:(j+50)<ny-1])
```

There might be a way with histogram and it's reverse index (possibly using ORD beforehand if data is float), but depending on the data (mostly how many different values end up being in data_max and how many values larger than it's smallest one exist) it might not be faster.

You can also try looping twice over -50 to 50, but i doubt that would be faster.

Good luck, Markus
